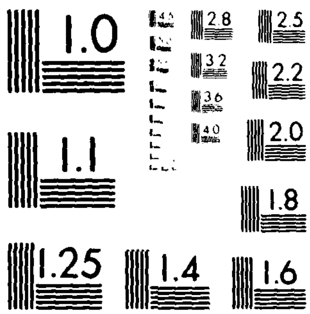
AD-A084 902 NAVAL POSTGRADUATE SCHOOL MONTEREY CA F/G 17/2
A STUDY OF THE COMMUNICATION CAPABILITIES OF THE OPARS FLIGHT P--ETC(U)
MAR 80 K L SMITH
UNCLASSIFIED NL

1 of 1
AD
A084902

END
DATE
FILMED
7-80
DTIC



MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS-1963-A

LEVEL II

(2)

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

6 A STUDY OF THE COMMUNICATION CAPABILITIES OF THE OPARS FLIGHT PLANNING SYSTEM FOR VARIOUS LEVELS OF DEMAND

by

10 Kenneth L. Smith

11 Mar[redacted]80 12 53

Thesis Advisor: Donald P. Gaver

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A084902	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A STUDY OF THE COMMUNICATION CAPABILITIES OF THE OPARS FLIGHT PLANNING SYSTEM FOR VARIOUS LEVELS OF DEMAND		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis March 1980
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Kenneth L. Smith		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE March 1980
		13. NUMBER OF PAGES 52
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

OPARS

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Through the Optimum Path Aircraft Routing System (OPARS), Fleet Numerical Oceanography center (FNOC) has committed itself to providing a computerized flight planning service remotely accessible via dial-up communications lines. The question arises as to whether the proposed number of telephone lines will be adequate to provide a level of service previously provided by the Lockheed Jetplan system. This study provides a detailed analysis of the response delays for the OPARS flight plan system. In addition,

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
(Page 1) S/N 0102-014-6601

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

estimates are given of communication requirements when various levels of demand prevail, and under conditions in which the FNOC computer is busy or idle.

Accession For	
NTIS GRA&I	✓
DDC TAB	
Unannounced	
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or special
A	

Approved for public release; distribution unlimited

A STUDY OF THE COMMUNICATION CAPABILITIES OF THE OPARS FLIGHT PLANNING SYSTEM FOR VARIOUS LEVELS OF DEMAND

by

Kenneth L. Smith
Lieutenant, United States Navy
B.S., The University of Pittsburgh, 1973

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
March 1980

Author: Kenneth L. Smith

Approved by: Donald P. Gaver
Thesis Advisor

Ellen F. Roland
Second Reader

Chairman, Department of Operations Research

Dean of Information and Policy Sciences

ABSTRACT

Through the Optimum Path Aircraft Routing System (OPARS), Fleet Numerical Oceanography Center (FNOC) has committed itself to providing a computerized flight planning service remotely accessible via dial-up communications lines. The question arises as to whether the proposed number of telephone lines will be adequate to provide a level of service previously provided by the Lockheed Jetplan system. This study provides a detailed analysis of the response delays for the OPARS flight plan system. In addition, estimates are given of communication requirements when various levels of demand prevail, and under conditions in which the FNOC computer is busy or idle.

TABLE OF CONTENTS

LIST OF FIGURES

NOTATIONS AND ABBREVIATIONS

A/B/m	m-Server queueing system where A represents the the interarrival distribution and B represents the service distribution
CM	Central Memory
CPU	Central Processor Unit
E_r	Denotes r stage Erlang distribution
E[T(Setup)]	Expected service time of IRG sub-system
E[T(Input queue)]	Expected service time of Input queue sub-system
E[T(Run)]	Expected service time of execute sub-system
E[T(Service)]	Expecter service time for OPARS system
FNOC	Fleet Numerical Oceanography Center
G	Denotes general distribution
λ_o	Arrival rate of OPARS jobs
λ_H	Arrival rate of higher priority jobs
μ	Service rate
M	Denotes Exponential distribution
N(t)	Number of customers in the system at time t
$\overline{N}$	Long run average number of customers in the system
OPARS	Optimum Path Aircraft Routing System
p_i	Erlang formula state probabilities for states $i = 1,2, \ldots ,11$
ρ	utilization factor

I. SUMMARY

In the proposed OPARS flight plan system, users at 37 remote terminals (Figure 1) connect directly to the Fleet Numerical Oceanography Center (FNOC) computer system via 11 telephone lines. The total amount of time that a line is busy, the total OPARS service time, can be modeled as the sum of three sub-system service times:

1. An interactive program setup time T(setup);
2. A delay that the OPARS program experiences in the FNOC computer's input queue T(input queue);
3. An OPARS program run time T(run).

The expected values of these individual sub-system service times can be computed and then summed to provide a mean or expected service time for the entire OPARS request as follows:

$$E[T(\text{service})] = E[T(\text{setup})] + E[T(\text{input queue})] + E[T(\text{run})].$$

From this, Erlang's formula equation (1) can be employed to compute individual long-run state probabilities (the probability that i lines are busy), p_i, and ultimately the probability, p_{11}, of having all eleven lines busy.

In support of the analytic model, a computer program was written to simulate the entire OPARS flight plan request process, from initial dial-up through program completion. The simulation also includes the arrival and servicing of other FNOC computer programs which interfere with the processing of the OPARS program.

Figure 1. Remote Terminal Sites

MOFFET NAS (2)	PATUXENT RIVER
ALAMEDA NAS (2)	MEMPHIS
JACKSONVILLE NAS (2)	CECIL FIELD
BRUNSWICK NAS	CHERRY POINT
NORTH ISLAND NAS (3)	EL TORO
NORFOLK (3)	WASHINGTON DC (FAA)
BARBERS POINT NAS (2)	ELIZABETH CITY (C.G.)
CHANUTE WY	KODIAK (C.G.)
NEW ORLEANS	SACRAMENTO (C.G.)
DETROIT	ST. PETERSBERG (C.G.)
SOUTH WEYMOUTH	LITTLE ROCK (C.G.)
WILLOW GROVE	MOBILE (C.G.)
GLENVIEW	WASHINGTON DC (C.G.)
WHIDBY IS.	BARBERS PT (C.G.)
POINT MAGU	

C.G. Coast Guard Station

Both the analytic model and the simulation, computed state probabilities for OPARS demand rates of 2, 4, 6, ...20 per hour and under conditions in which the FNOC computer is busy or idle. The results clearly indicate that the 11 telephone lines can handle demand rates up to three times as great as those currently being experienced by the Lockheed Jetplan system before p_{11}, the probability of all lines busy, exceeds, .05.

II. INTRODUCTION

Fleet Numerical Oceanography Center (FNOC) is currently testing and evaluating a computerized flight plan system, referred to, for short, as OPARS. This sytem, developed to replace the Lockheed Jetplan flight plan sytem, provides users at remote sites with direct access to the FNOC computer via 11 telephone lines. The purpose of this study is to determine if the intended number of telephone lines would be adequate to ensure a low probability of having all lines busy.

The number of lines busy at any time t, $\{N(t) | t \geq 0\}$ can be modeled as a Birth-Death process with inter-arrival times that are assumed to be independent and exponentially distributed but with service times that clearly are not. Fortunately, in a communications system characterized by calls that arrive in a stationary or time-homogenous Poisson Process of rate λ and vie for n lines with no queue (blocked calls lost), the long run probability of having i lines busy, p_i, can be computed from Erlang's formula.

$$p_i = \begin{cases} \dfrac{(\lambda\bar{t})^i / i!}{\sum\limits_{j=0}^{N} (\lambda\bar{t})/j!} & 0 \leq i \leq N \\ 0 & i > N \end{cases} \qquad (1)$$

This formula has the surprising feature of being valid for any service time distribution. Indeed, given the above assumption, one need only obtain the mean service time, $\bar{t}$,

in order to calculate p_i. The majority of the effort in this study was to characterize the OPARS system in such a way so that a mean service time could be derived under various operating conditions.

This paper begins with some introductory material concerning the FNOC computer systems, the OPARS flight plan system and historical usage of the Lockheed Jetplan system. Section IV is a detailed description of the analytic model used and a discussion of the supporting siumulation program. Finally, Section V contains the tabled results and an analysis.

Notation used in this paper is consistent with that found in Queueing Theory literature and will be explained whenever introduced. In addition, a listing and discription of all notation and abbreviations can be found on page seven.

III. BACKGROUND

A. FNOC COMPUTER SYSTEM

FNOC currently has four computers in service at their Monterey facility, but only one, referred to as HAL is accessible to the remote OPARS terminals. HAL is a CDC-6600 computer with two central processors and 330 K of octal central memory. HAL is accessed within FNOC by approximately twenty-five interactive terminals as well as by a batch job system that routinely processes hundreds of programs a day. Since HAL alone is involved in providing service to OPARS remote terminals, we will only consider it in the discussion.

1. HAL Batch System

HAL's batch system is composed of a priority ranked "first-in, first-out" input queue, which can be considered exterior to the computer, and a priority ranked execute queue from which jobs are selected by the scheduler for processing (Figure 2). When a job enters the system, it first enters the input queue, at which the job's user-assigned priority establishes its initial position. While in the queue the job slowly accrues additional "wait time" priority which ensures that even the lowest priority jobs are eventually run. The largest number of daily jobs are restricted to priorities of 1, 2 or 3 (3 being the highest) but priorities up to 77 can be assigned. The priority of OPARS jobs is fixed automatically at 60 which immediately puts it ahead of all but a few other jobs in the queue.

When space is available within the computer the scheduler removes the first job (with respect to priority then to time-of-arrival) from the input queue and moves it to the execute queue. Upon entering the execute queue the job loses all of its previously gained "wait-time" priority but immediately begins to gain it back as the job waits to be processed. In this queue the jobs with the higher assigned priorities gain this additional priority faster than those with low assigned priority. This mechanism results in higher priority jobs getting selected for processing sooner than low priority jobs. When selected by the scheduling routine, the program is moved into central memory and it is processed for a unit length of time or until it attempts to access a device (disk, tape, etc.) which is unavailable. When this event occurs the job is removed form central memroy and returned to the execute queue, having its accured priority reset to zero. The process continues in this manner until the program is completed and it is transferred to the output queue.

2. Current FNOC Workload

From available central processor and central memory utilization profiles (Figure 3) it can be readily seen that resource demands on HAL can be separated into two states: A Busy or High Demand state in which the computer is operating at maximum capacity, and an Idle or Low Demand state in which most of the resources are immediately available. The High Demand state is characterized by a full execute queue, and a lengthy input queue, while the Low Demand state exhibits an

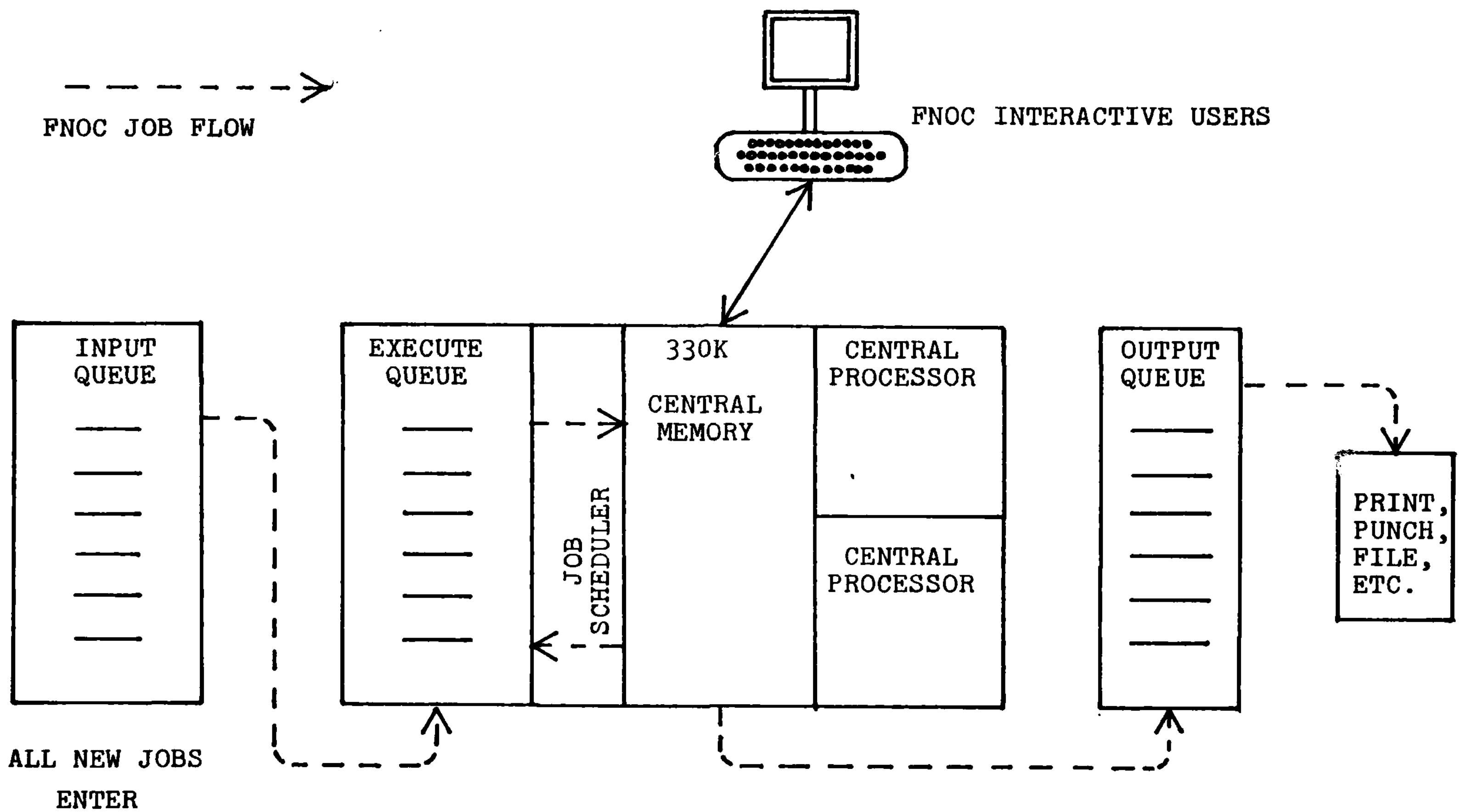


Figure 2. FNOC Computer System

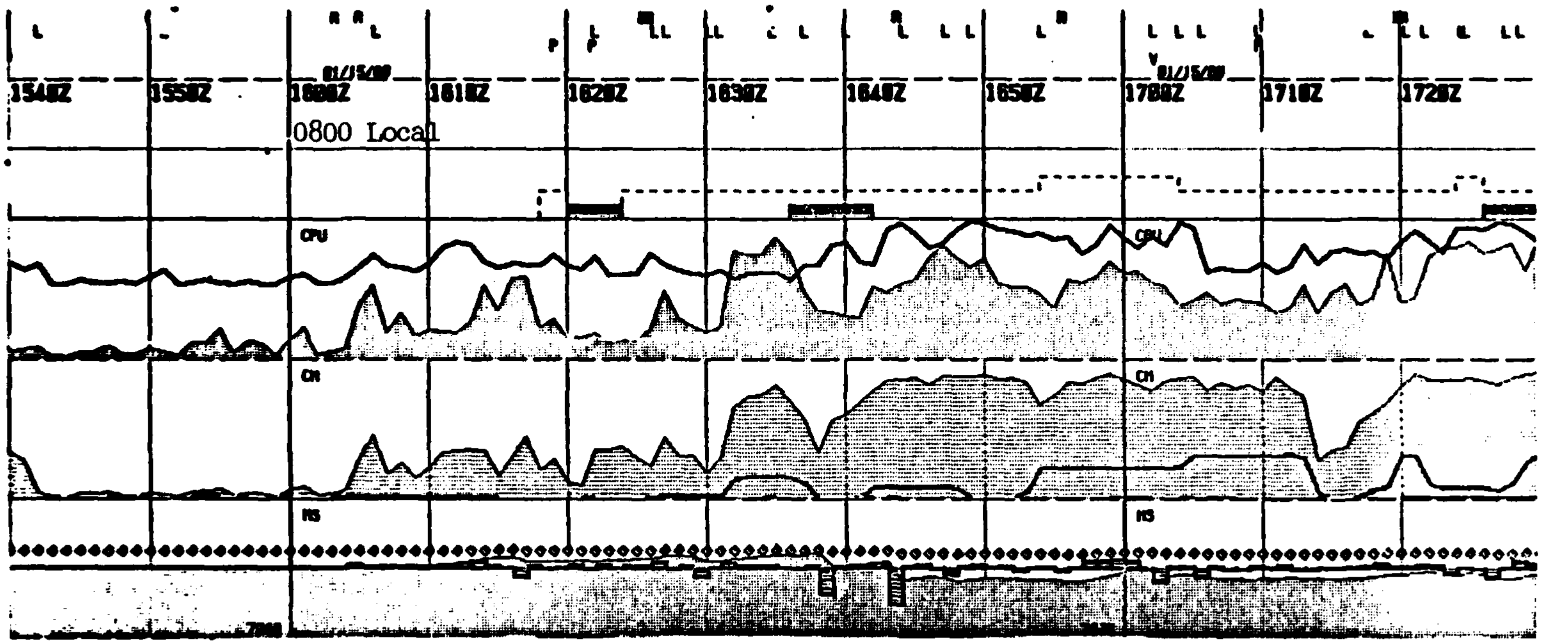
execute queue with only an occasional job in it and an input queue which is empty. Fortunately the transition period between these states is quite brief and therefore a transition state is not needed.

The High Demand period coincides, not surprisingly, with normal working hours except that it extends well into the evening, often as late as midnight. Any jobs entering the system during this period must be delayed for a time in the input queue before being run. During the idle period, on the other hand, jobs pause only momentarily in the input queue before being run.

B. OPARS FLIGHT PLAN SYSTEM

The OPARS flight plan system provides the user with an optimum (with respect to time) route of flight given forecast weather and user inputs such as aircraft type and take off weight. For flights within areas that have a defined routing structure the optimization is fairly easy because only a few routes are candidates for the optimum. Over water or point to point flight plans are more difficult because there is, in theory, an uncountably infinite number of available routes.

There are two major sections of the OPARS program: an interactive portion and the optimization program. When a user at a remote site requests an OPARS flight plan one first sets up the program with the Interactive Response Generator (IRG). Using a query and response technique, the IRG prepares the main program by obtaining required



CPU - Central Processor Unit

CM - Central Memory

Figure 3. CPU and CM Utilization Profiles

information from the user. During this session the IRG is not checking input information for validity, but only for format. For example, an entry of ABCE, as the four-letter identification code for the destination airfield, would be accepted even though no such airfield code exists. When all of the required information is entered, the IRG allows the user to review and change any input if necessary. Then, upon command, the program is put into the HAL computer batch system.

As discussed in the previous section, the OPARS job enters the input queue along with all other FNOC jobs and awaits its turn. Hereafter the OPARS program is handled as any other batch program with one exception. Upon completion the flight plan is returned automatically to the terminal where the request was entered. If for some reason the line has been disconnected, the flight plan is placed into an output file from which it can be retrieved later.

C. HISTORICAL USAGE/PROJECTED DEMAND

In an effort to evaluate expected demand for the OPARS flight plan, several methods were employed. They were:

1) A linear regression model of historical usage by the Navy to estimate demand through 1981;
2) expectations of remote site users;
3) Lockheed's records of previous use.

None of these methods alone provided the required level of accuracy and, only by combining the three, could a reasonable estimate be made.

The Navy's own records provided the best overall information. This information, shown in Figure 4, consists of monthly totals of Navy requests for the Jetplan. The trend is clearly an increase in usage and a simple linear regression yielded the projected demands shown in Figure 5. These monthly totals, unfortunately did not provide any information as to how requests varied during the day.

	1977 Month(per day)	1978 Month(per day)	1979 Month(per day)
JAN	866(28.6)	1177(38)	2033(65.6)
FEB	1012(34.9)	1107(39.5)	1926(68.8)
MAR	1076(34.7)	1856(59.9)	2107(68)
APR	1169(38.9)	1319(44)	1887(62.9)
MAY	1103(35.6)	1490(48.1)	2119(68.4)
JUN	1174(39.1)	1224(40.8)	2081(69.4)
JUL	1711(55.2)	1146(37)	2011(64.9)
AUG	1066(34.4)	1365(44)	2247(72.5)
SEP	977(32.3)	1226(40.9)	2218(73.9)
OCT	1142(36.8)	1415(45.6)	
NOV	1103(36.8)	1321(44)	
DEC	1107(35.7)	1386(44.7)	

Figure 4. Historical Usage of Lockheed's Jetplan

Y INTERCEPT 28.79	JUL 80 ESTIMATE 80.4 /DAY
SLOPE 1.2	JUL 81 ESTIMATE 94.8 /DAY
COR. COEFF. .826	

Figure 5. Linear Regression of Usage Data

In order to obtain the distribution of daily requests, expected demand information was requested from all remote site users. The resulting information suffered from imprecision and a lack of completeness (only about half of the remote sites responded).

As a final effort, an attempt was made to secure actual daily records from the Lockheed Corporation. The Lockheed personnel were unable to provide me with actual data, but over the course of several interviews a picture of the daily demand distribution was pieced together.

The results of these data collection efforts appear in Figure 7. These graphs show the projected demand for July 1980 and July 1981 distributed using the composite results of Lockheed information and user expectations.

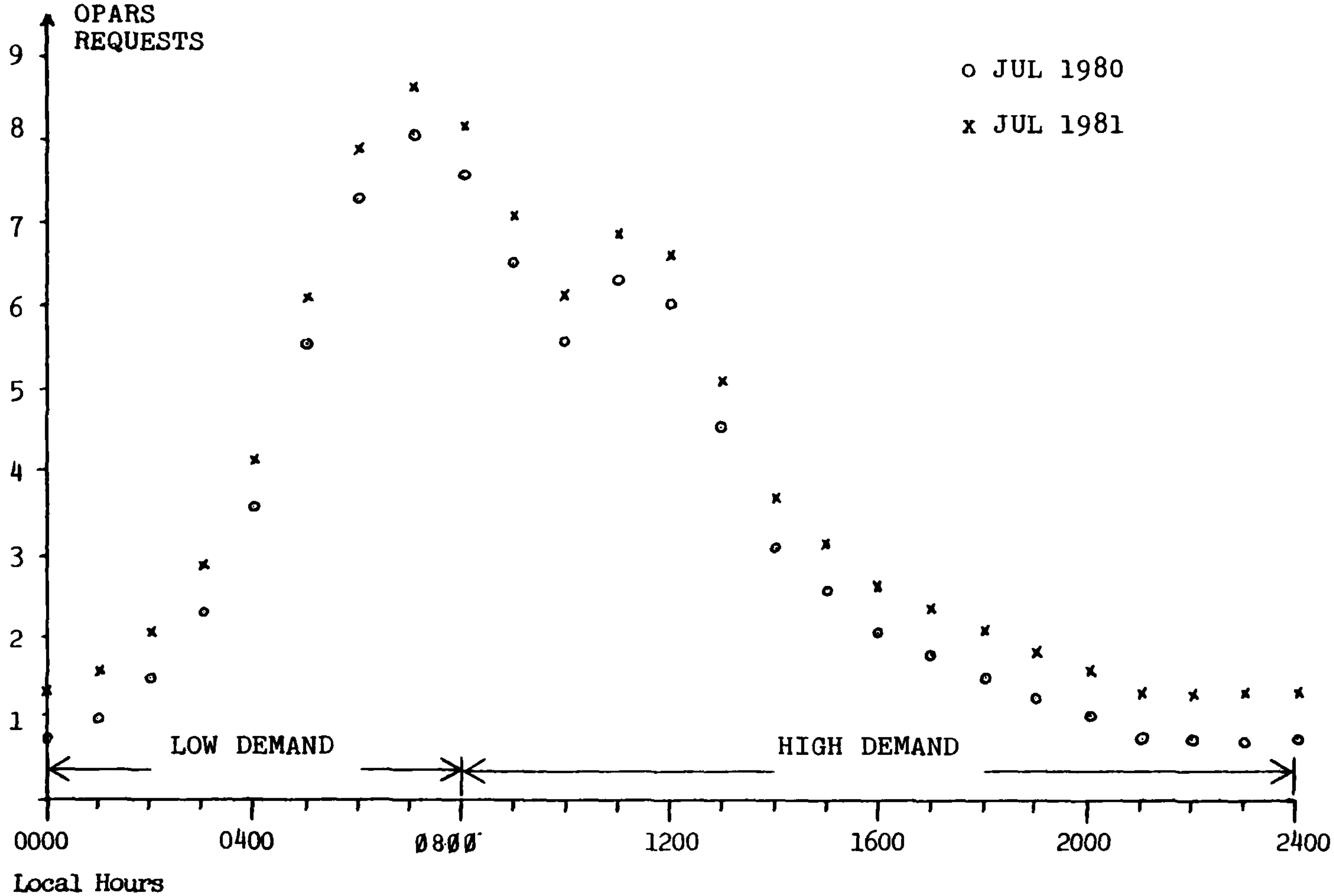


Figure 6. Predicted Demand Rates for OPARS

IV. THE MODEL

We start the modeling discussion by reviewing some basic assumptions about the system.

First and foremost is the assumption concerning time-homogeneous Poisson arrivals for OPARS requests. Without this assumption, the Erlang formula (1) would not be valid and a much more complex model would be required.

Secondly, the service times for the three sub-systems are assumed to be mutually independent.

Finally, as discussed in the last section, the conditions for the computers workload will be divided into two distinct periods. These are the High Demand period with a nonempty input queue, and the Low Demand period where the OPARS requests bypass the input queue and go directly into the computer.

A. THE ANALYTIC MODEL

Since the main thrust of the analytic effort is to obtain an overall mean service time, an immediate simplification would be to somehow break the OPARS system into two or more sub-systems whose service times would be more easily obtained. With this in mind, we start by separating the service into the interactive sub-system and the batch sub-system. The latter is still rather formidable because of the nature of the input and execute queues and the complexity of the scheduling routine. The final step is to reduce the batch system

into an input queue sub-system and an execution sub-system. The reasons for this may not be readily apparent as there are other modeling techniques such as an M|G|m multiserver queueing system with priorities, which would handle the system as a whole. Such a system would require that each job priority have a mean service time. Unfortunately, there is no correlation between a jobs priority and its core and central processor requirements. Additional unrealistic assumptions would be required which would, in the end, reduce the validity of the result.

1. The Interactive Sub-System

First to be considered is the interactive sub-system. An estimate for the mean time for this sub-system is readily comput ble from the available, albeit scarce, data. This data(Figure 7) was obtained by timing FNOC technicians on trial OPARS runs. The mean of these data is 5.1 minutes and will be assumed to be the same during both High and Low Demand periods.

2. Input Queue Sub-System

The next sub-system to model is the input queue sub-system. Under low demand conditions the mean service time is, by definition, equal to zero. During the High Demand period it is not quite so simple. Again, by previous definition, the High Demand period has a continuously non-empty input queue. This very simple assumption suggests that the input queue may be modeled as a single-server queueing system by itself, the server being the first position in the queue. One must merely postulate the interarrival process

4'9"	7'22"
6'2"	4'37"
5'09"	5'17"
5'01"	4'21"
3'57"	4'55"

$\overline{X}$ = 5.1 MIN

σ = 1.0

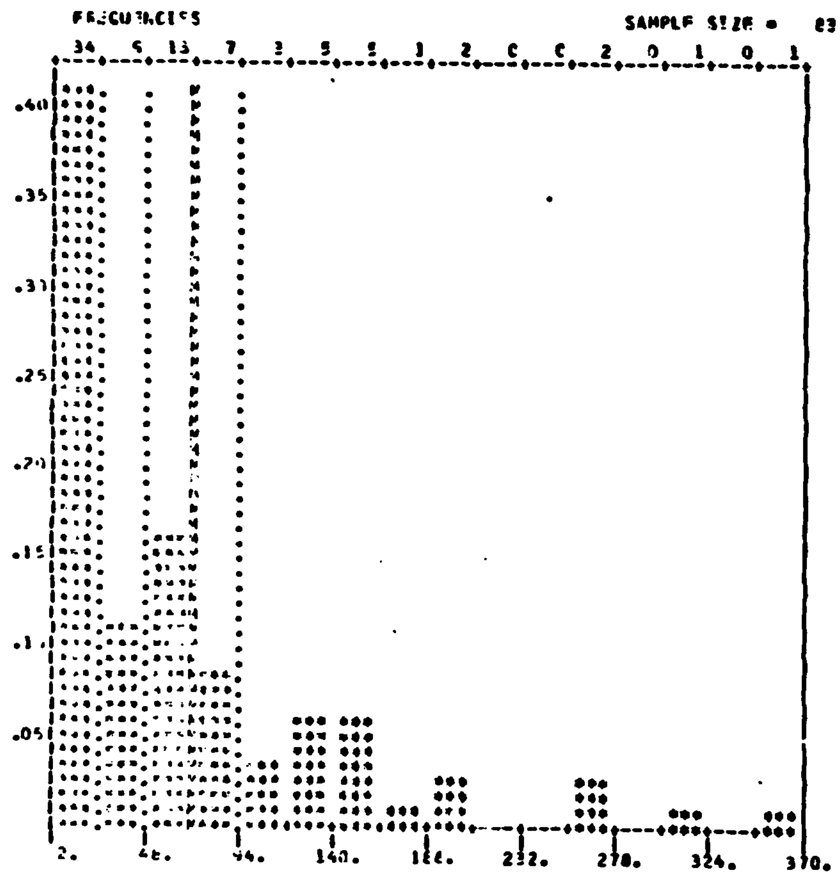
Figure 7. Interactive Setup Times

and the service process, the latter representing the times between successive departures from the input queue. If we can now model the arrival rate as Poisson and the service distribution as exponential we have an M/M/1 queueing system ranked by priority for which there is a simple closed form solution for expected waiting times. The interested reader is urged to consult Griffin [2] for the derivation of this result.

There was no data available to determine the precise manner in which OPARS jobs enter the input queue and so a Poisson arrival rate is as plausible as any other. However the service distribution (the interdeparture times from the input queue) can be measured by observing a real time display of HAL's input queue and timing departures. The resulting data (Figure 8) and histogram (Figure 9) are surprisingly close to the exponential distribution. In fact, the Chi-squared goodness of fit test resulted in a test statistic of

46.	41.	20.	154.
77.	3.	9.	155.
35.	4.	16.	40.
94.	81.	370.	10.
21.	105.	193.	53.
10.	58.	2.	21.
12.	174.	123.	60.
139.	155.	32.	24.
25.	20.	101.	274.
33.	16.	8.	49.
12.	24.	30.	2.
23.	17.	10.	193.
51.	8.	73.	62.
55.	260.	88.	113.
145.	138.	30.	3.
65.	119.	25.	8.
5.	20.	9.	25.
52.	93.	303.	75.
18.	61.	125.	29.
54.	17.	3.	10.
145.	55.	65.	

Figure 8. Input Queue Interdeparture Time



CENTRAL TENDENCY		SPREAD		HIGHER CENTRAL MOMENTS		DISTRIBUTION		
MEAN	[illegible]	VARIANCE	5.3982548E 03	M3	7.5642499E 05	MINIMUM		2.0000000E 00
MEDIAN	[illegible]	STD DEV	7.3472818E 01	M4	2.3025226E 08	.10 QUANTILE		8.0000000E 00
TRIMEAN	[illegible]	COEF VAR	1.0526735E 00	SKEWNESS	1.9071559E 00	.25 QUANTILE	(HINGE)	[illegible]
MIDMEAN	[illegible]	MEAN DEV	5.0144588E 01	KURTOSIS	3.8731776E 00	.50 QUANTILE	(MEDIAN)	4.1000000E 01
MIDRANGE	[illegible]	RANGE	3.6800000E 02	BETA1	7.2930435E 05	.75 QUANTILE	(HINGE)	5.4000000E 01
GEOM MEAN	[illegible]	MIDSPREAD	[illegible]	BETA2	1.5270560E 08	.90 QUANTILE		1.6420000E 02
HARM MEAN	[illegible]					MAXIMUM		3.7000000E 02

Figure 9. Histogram of Input Queue Interdeparture Time Data

5	0	6	1	6
1	3	0	2	5
7	3	1	9	2
0	3	4	0	0
14	1	0	6	0
5	5	3	3	7
1	4	1	1	2
0	1	0	2	1
2	19	8	5	0
6	5	4	5	

$$\lambda_H = \frac{1}{\bar{x}} = 3.45/HR$$

Figure 10. Higher Priority Job Counts

0.533, which indicates that the data is very close to being exponential with $\lambda_0 = 53.52$ per hour.

The final consideration is the priority ranking. As stated earlier an OPARS job is assigned a priority of 60 while FNOC jobs can have priorities from 1 to 77. However, the batch jobs of lower priority can be ignored and we are left with only higher priority jobs to consider. Available data (Figure 10) showed higher priority jobs arriving at the rate of 3.45 per hour. The one remaining assumption is that these higher priority jobs also arrive according to a Poisson Process.

The expression for OPARS expected input queue is then,

$$E[T(\text{INPUT QUEUE})] = \frac{1}{\mu\left(1 - \frac{\lambda_O + \lambda_H}{\mu}\left(1 - \frac{\lambda_H}{\mu}\right)\right)} \quad (2)$$

where μ is the service rate, μ is the OPARS arrival rate and λ_H is the higher priority job arrival rate. This expression is valid only when:

$$\frac{\lambda_H + \lambda_\mu}{\mu} > 1.$$

This input queue waiting time was computed for OPARS arrival rates of 2, 4, 6,...,20 per hour with results listed in column three of Figure 16.

3. The Execution Sub-Systems

In the third and final section, the execution sub-system, we will characterize the OPARS program run time as well as delays due to other factors.

Because of the central memory requirements of the OPARS program (150K) only two OPARS jobs are allowed in the computer at one time. Any additional requests arriving during such a state would be required to wait in the input queue and would be passed over by the scheduling routine as it selected jobs for processing. Because the two OPARS jobs can be processed simultaneously, a queueing system of the form GI/G/2 is suggested. This type of system is uncomfortable to work with, so simplifying assumptions will be made.

First, because of a lack of data to the contrary, we will again ssume Poisson arrivals of OPARS jobs. It now remains to describe the service distribution. Unfortunately, even with Poisson arrivals, multi-server systems with any but exponential service times, are difficult to analyze mathematically.

Figures 11 and 12 contain actual OPARS run-time data for Low Demand and High Demand periods respectively. These data reflect the time an OPARS job spends in the system from it's transfer to the execute queue until it's completion. Histograms of these data are in Figures 13 and 14. and are clearly not exponential . However, if the data can be considered to be from an Erlang distribution, then the nomograph in Figure 15 (see Hillier and Leiberman [4]) can be used to obtain an approximate value for $\bar{N}$, the mean number of OPARS jobs in the system. From this the mean service time can be computed via Little's result:

$$E[T(run)] = \frac{\bar{N}}{\lambda_o} \qquad (3)$$

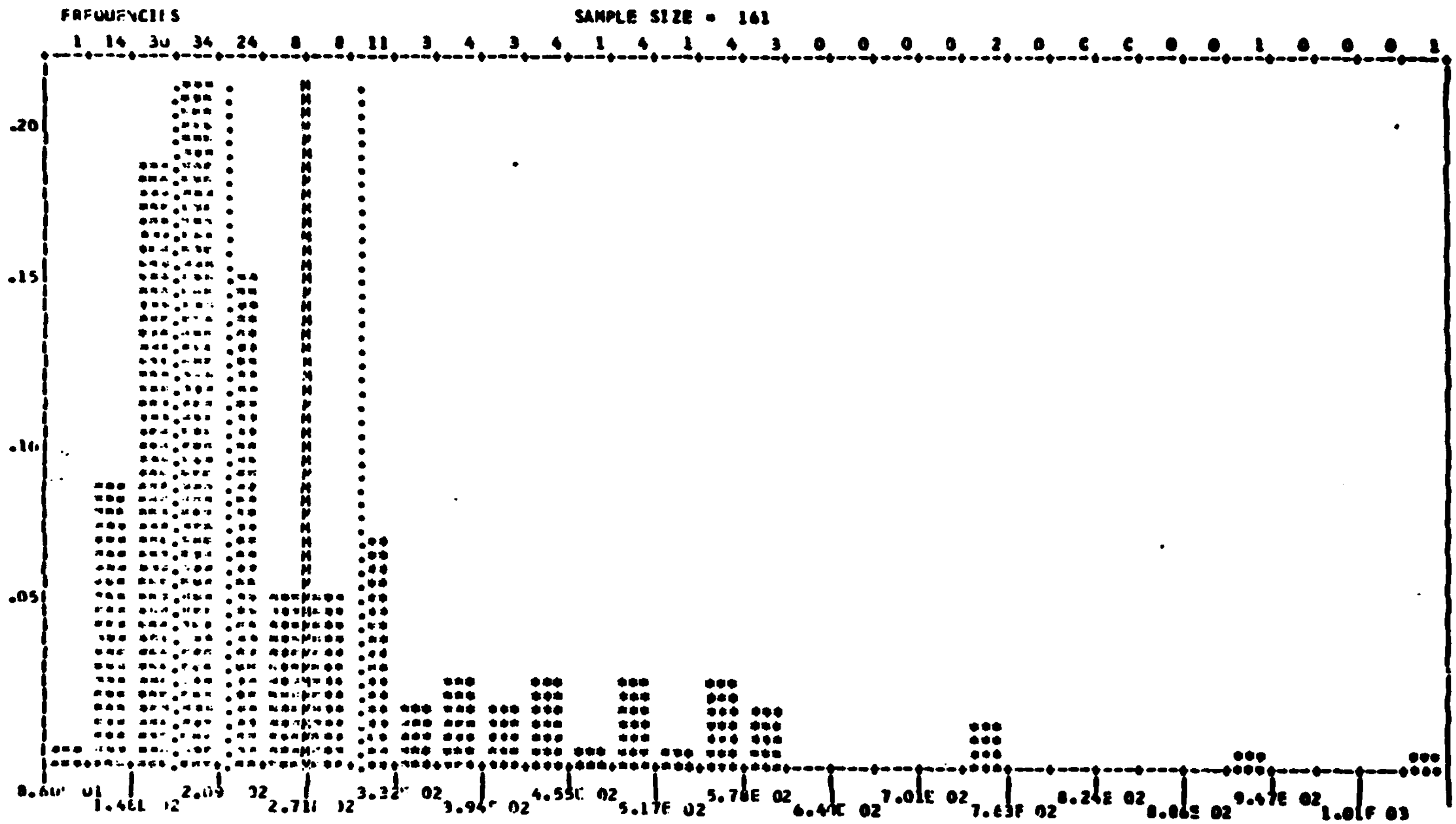
where λ_o is, as before, the arrival rate for OPARS jobs.

305	184	162	315	166	171
146	205	267	228	171	224
178	241	211	210	182	217
248	129	172	235	178	161
592	196	178	165	167	504
189	199	194	158	207	420
86	201	551	170	483	433
393	229	274	161	272	301
338	231	180	145	316	277
270	236	581	184	184	591
206	230	160	374	177	197
305	222	310	141	147	184
348	232	141	320	141	159
185	197	298	339	311	237
231	195	495	171	200	165
189	221	271	227	195	547
412	233	211	304	188	179
163	755	182	437	152	164
310	750	164	573	289	197
195	304	186	211	374	188
280	561	140	236	199	243
391	242	175	267	161	194
163	201	126	231	415	512
218	194	130	297	303	493
168	210	134	168	219	139
200	251	928	173	449	174
1070	187	133	425	146	

Figure 11. OPARS Program Run Times: Low Demand

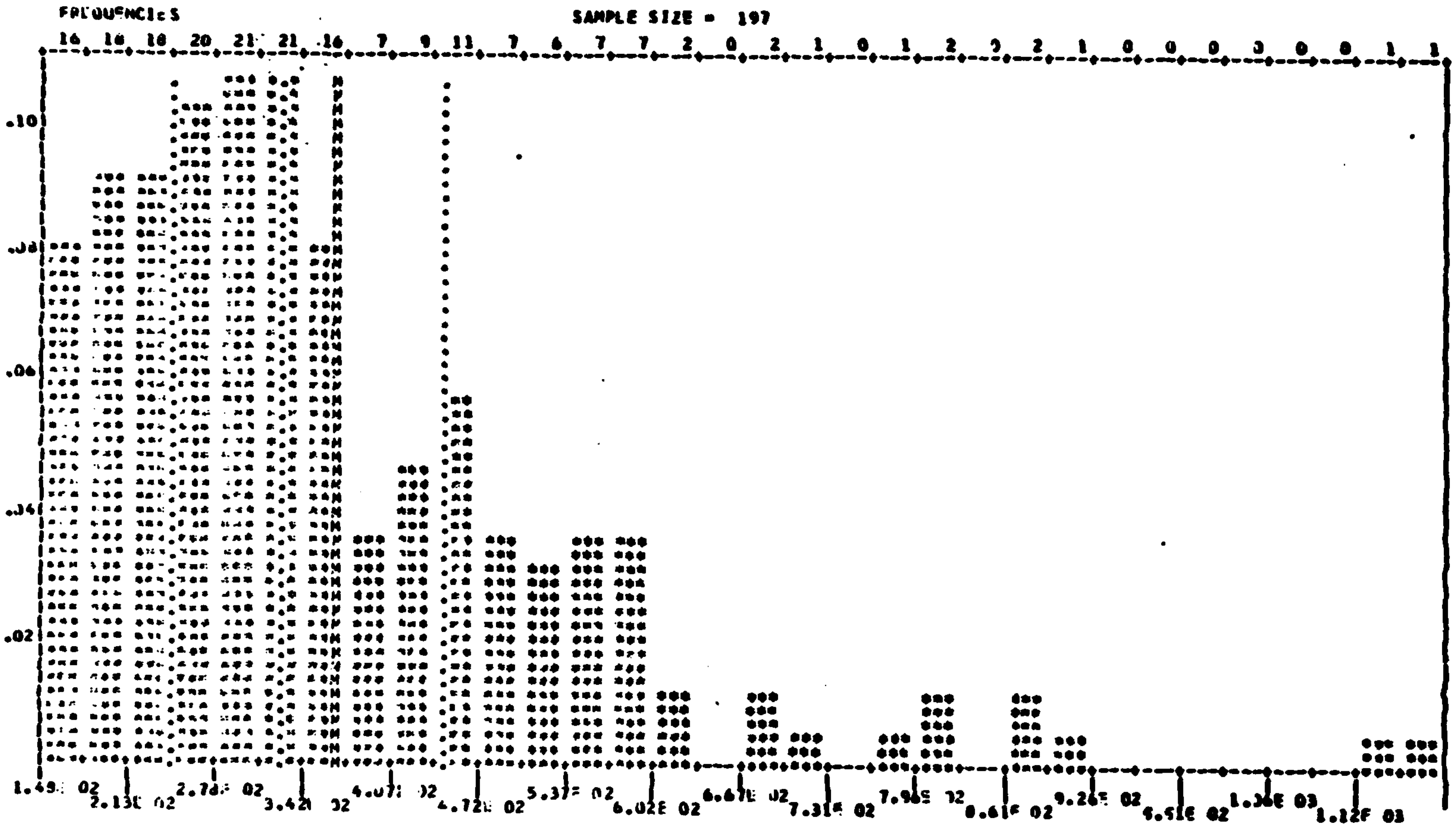
258	295	192	261	391	578	228	356
261	272	320	165	232	279	208	392
222	313	221	170	241	299	920	241
148	598	472	190	207	1122	511	222
180	326	339	280	295	286	367	314
291	331	591	249	563	268	719	175
545	516	512	335	333	368	341	207
272	482	442	162	266	273	274	295
321	789	544	211	351	491	313	563
388	358	433	191	882	267	344	333
336	363	279	182	484	249	333	266
420	346	249	197	298	455	377	351
200	173	169	231	206	804	272	881
242	528	160	282	349	211	236	484
283	283	200	536	485	538	160	298
442	286	363	455	352	421	322	286
355	220	171	179	334	217	260	285
340	395	417	609	456	578	682	437
248	683	472	348	369	389	478	278
228	808	210	400	626	601	447	238
543	326	190	284	426	235	594	410
340	412	215	160	581	209	240	300
1185	469	168	468	520	210	244	
461	195	164	371	539	335	154	
305	248	272	502	436	250	332	

Figure 12. OPARS Program Run Times: High Demand



CENTRAL TENDENCY		SPREAD		HIGHER CENTRAL MOMENTS		DISTRIBUTION	
MEAN	2.654874[illegible] 02	VARIANCE	2.275760E 04	M3	8.177864[illegible] 06	MINIMUM	8.600000E 01
MEDIAN	[illegible] 02	STD DEV	1.508562E 02	M4	5.221028[illegible] 09	.10 QUANTILE	1.520000E 02
TRIMEAN	2.246251[illegible] 02	COEFF VAR	5.675396E-01	SKEWNESS	2.382046[illegible] 00	.25 QUANTILE (HINGE)	[illegible]
MIDMEAN	[illegible] 02	MEAN DEV	5.315875[illegible] 01	KURTOSIS	7.080990[illegible] 00	.50 QUANTILE (MEDIAN)	[illegible]
[illegible]	[illegible]	RANGE	9.940000E 02	BETA1	8.026115[illegible] 06	.75 QUANTILE (HINGE)	[illegible]
GEOM MEAN	2.377796[illegible] 02	MIDSPREAD	1.295000E 02	BETA2	5.110444[illegible] 09	.90 QUANTILE	4.490000E 02
HARM MEAN	2.136313[illegible] 02					MAXIMUM	1.080000E 03

Figure 13. Histogram of OPARS Program Run time Data: Low Demand



CENTRAL TENDENCY		SPREAD		HIGHER CENTRAL MOMENTS		DISTRIBUTION	
MEAN	3.611318E 02	VARIANCE	3.004467E 04	M3	9.135643E 06	MINIMUM	1.482233E 02
MEDIAN	3.22[illegible] 02	STD DEV	1.733340E 02	M4	6.585623E 09	.12 QUANTILE	1.[illegible] 02
TRIMEAN	3.323750E 02	COEFF VAR	4.799742E-01	SKEWNESS	1.755004E 00	.25 QUANTILE (HINGE)	2.413000E 02
MIDMEAN	3.241766E 02	MEAN DEV	1.227766E 02	KURTOSIS	4.295616E 00	.51 QUANTILE (MEDIAN)	3.220000E 02
MIDRANGE	6.665[illegible] 02	RANGE	1.037000E 03	BETA1	9.070933E 06	.75 QUANTILE (HINGE)	4.445000E 02
GEOM MEAN	3.2[illegible]4646E 02	MIDSPREAD	2.035000E 02	BETA2	6.475156E 09	.97 QUANTILE	5.63[illegible] 02
HARM MEAN	3.[illegible]9143E 02					MAXIMUM	1.185000E 03

Figure 14. Histogram of OPARS Program Run Time Data: High Demand

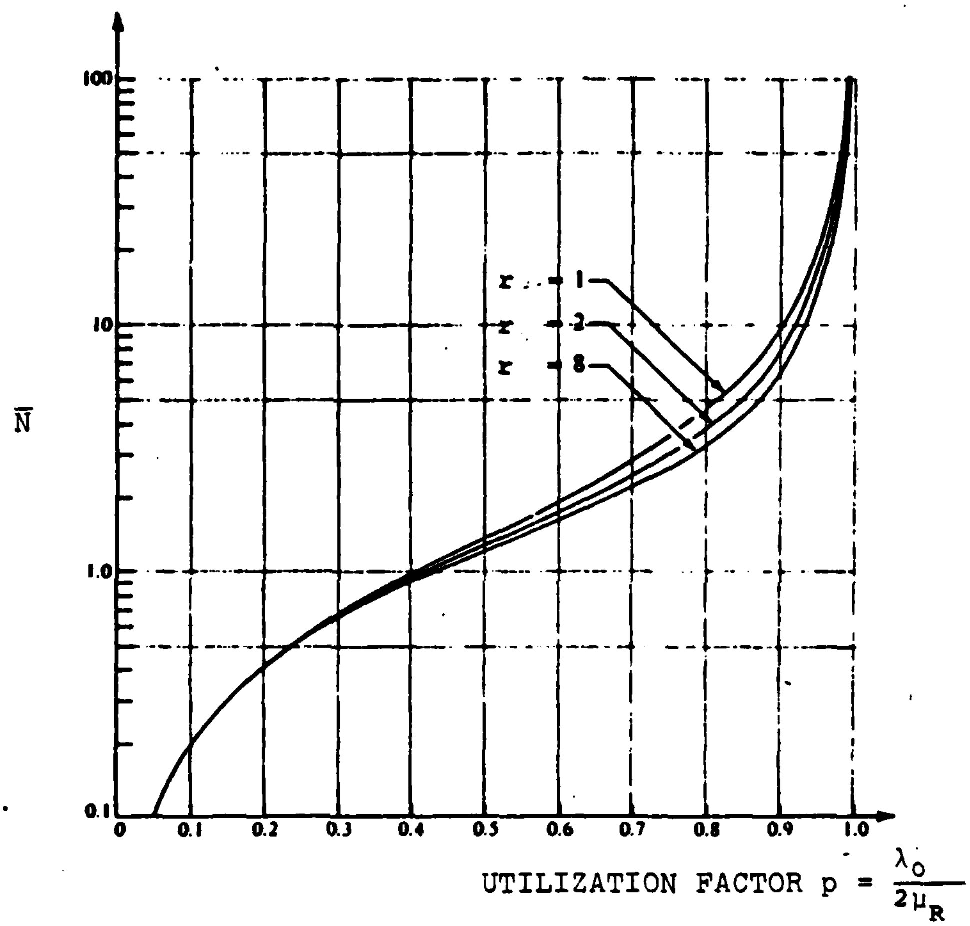


Figure 15. Nomograph of the Long Run Average Number of Customers, $\overline{N}$, in an $M/E_r/2$ Queue

λ_o	E[T(setup)]	E[T(input)]	$\rho/\overline{N}$/E[T(run)]	E [T(services)]
2/HR	5.1	1.33	.1/.2/6.0	12.43
4/HR	5.1	1.39	.2/.4/6.0	12.49
6/HR	5.1	1.45	.3/.64/6.4	12.95
8/HR	5.1	1.52	.4/.94/7.1	13.72
10/HR	5.1	1.60	.5/1.3/7.8	14.5
12/HR	5.1	1.68	.6/1.7/8.5	15.28
14/HR	5.1	1.78	.7/2.4/10.3	17.18
16/HR	5.1	1.88	.8/3.8/14.3	21.28
18/HR	5.1	1.99	.9/7.6/25.3	32.39
20/HR	5.1	2.13	1.0/∞ / ∞	∞

Figure 16 . Busy System Summary (Time in Minutes)

λ_o	E[T(setup)]	E[T(input queue)]	ρ / $\overline{N}$ / E[T(run)]	E[T(Services)]
2/HR	5.1	0	.074/.15/4.5	9.6
4/HR	5.1	0	.148/.30/4.5	9.6
6/HR	5.1	0	.222/.46/4.6	9.7
8/HR	5.1	0	.296/.63/4.7	9.8
10/HR	5.1	0	.369/.83/5.0	10.1
12/HR	5.1	0	.443/1.1/5.5	10/6
14/HR	5.1	0	.517/1.4/6.0	11.1
16/HR	5.1	0	.591/1.7/6.4	11.5
18/HR	5.1	0	.665/2.1/7.0	12.1
20/HR	5.1	0	.738/2.8/8.4	13.5

Figure 17. Idle System Summary (Time in Minutes)

The run data was parameterized as an Erlang distribution with the following results:

	BUSY	IDLE
r	4	3
λ	.6645	.6772
$\frac{1}{\mu_R} = \frac{r}{\lambda}$	6.02 min	4.43 min

Figure 18. Erlang Distribution Parameters

Where $\frac{1}{\mu_R}$ is the mean run time.

Finally, the expected execution service time, E[T(runs)], was computed for a series of OPARS arrivals, as before, with the results in column 4 of Figures 16 and 17.

4. Summary

In summary we have now succeeded in representing the total OPARS service time as the sum of three, readily computable sub-system times (Figure 19):

$$E[T(\text{service})] = E[T(\text{setup})] + E[T(\text{input queue})] + E[T(\text{run})]. \quad (4)$$

Erlang's formula can now be employed to compute expected state probabilities for lines busy.

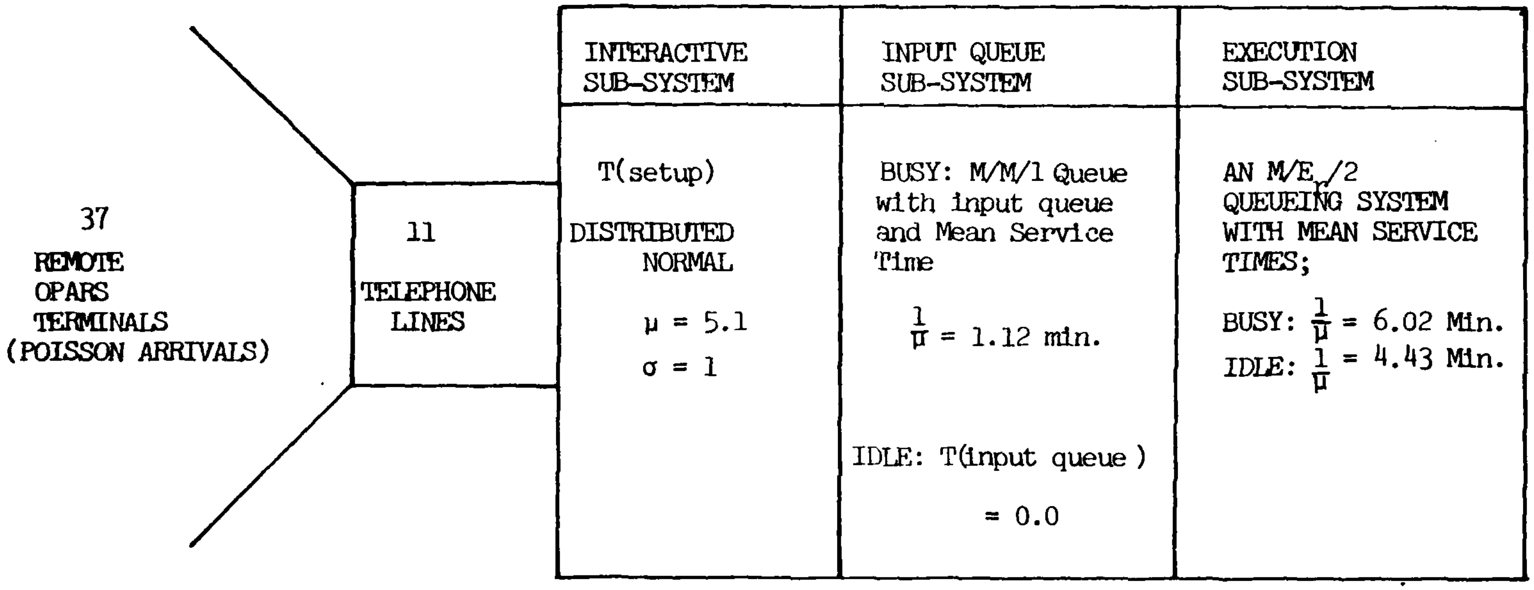


Figure 19. OPARS System Diagram

B. THE SIMULATION

As an alternative solution technique, two computer programs were written to simulate the OPARS system, one each for High and Low Demand computer states. Output of the simulations were, as in the analytic model, state probabilities for lines busy under various demand states.

There are two key differences in the logic flow for the two simulations. First OPARS requests arriving during the busy state are sent to the input queue where they must wait to be served. Under idle conditions jobs go directly to the computer. Secondly, in the busy state, higher priority jobs are being created which interfere with the processing of OPARS jobs. The following is a list of the various distributions and their parametric values used in the simulation.

OPARS Interarrival time: EXP (λ= 2,4,6,...20/HR)

High Priority Jobs Interarrival time: EXP(λ= 3.45/HR)

IRG service times: NORMAL (μ= 5.1 Min., σ= 1)

Input queue Interdeparture times: EXP(λ= 53.5/HR)

OPARS run times, Idle: GAMMA (λ= .721, r = 4.34)

OPARS run times, Busy: GAMMA (λ= .701, r = 3.1)

Figure 20: Probability Distribution Used in Simulation Program

The simulations were run for 48 hours for each OPARS demand rate and under each computer state.

The only significant factor included in the simulations but not in the analytic model had to do with a retrial population. In the event of all lines busy, the analytic model assumes arriving requests disappear, but in reality, people, when faced with a busy signal, hang up and try again. The simulation retries all balked attempts after ten minutes.

V. RESULTS AND ANALYSIS

Erlang formula (1) state probabilities, pi, were computed for OPARS request rates of 2,4,6,...,20 per hour under High and Low Demand conditions. Appendix A contains the FORTRAN program used to calculate these state probabilities given demand rate and mean service time, E[T(service)]. Similarly Appendices D and C contain the SIMSCRIPT programs which simulate the OPARS system.

The results of both methods are in Figures 2. and 22. The state probabilities are suprisingly close (differing by less than 2% in most cases) and clearly show that with eleven operating telephone lines, demand rates must be far in excess of projected requirements (Section III.C), before the probability of all lines being busy is of concern.

One should be aware, however, that in spite of the closeness of the results these two solutions merely serve to verify each other. Because the OPARS system is not fully functional at the time of this report, there is no way to validate either method against the actual system.

In addition, if any significant changes are made to either the OPARS system or to the FNOC computer system, a new evaluation would be required.

HIGH DEMAND PERIOD

STATE	2/HR	4/HR	6/HR	8/HR	10/HR	12/HR	14/HR	16/HR	18/HR	20/HR
0	0.6608	0.4349	0.2739	0.1605	0.0892	0.0471	0.0182	0.0035	0.0001	0.0000
1	0.2738	0.3621	0.3547	0.2936	0.2156	0.1439	0.0728	0.0198	0.0008	0.0000
2	0.0567	0.1508	0.2297	0.2686	0.2605	0.2199	0.1460	0.0561	0.0039	0.0000
3	0.0078	0.0418	0.0991	0.1638	0.2099	0.2239	0.1951	0.1060	0.0126	0.0001
4	0.0008	0.0087	0.0321	0.0749	0.1268	0.1711	0.1955	0.1504	0.0307	0.0006
5	0.0000	0.0015	0.0083	0.0274	0.0613	0.1046	0.1568	0.1706	0.0597	0.0026
6	0.0000	0.0002	0.0018	0.0022	0.0247	0.0533	0.1048	0.1614	0.0966	0.0086
7	0.0000	0.0000	0.0003	0.0005	0.0085	0.0232	0.0600	0.1308	0.1342	0.0245
8	0.0000	0.0000	0.0000	0.0001	0.0026	0.0089	0.0301	0.0928	0.1630	0.0612
9	0.0000	0.0000	010000	0.0000	0.0007	0.0030	0.0134	0.0585	0.1761	0.1360
10	0.0000	0.0000	0.0000	0.0000	0.0002	0.0009	0.0054	0.0332	0.1711	0.2720
11	0.0000	0.0000	0.0000	0.0000	0.0000	0.0003	0.0020	0.0171	0.1512	0.4945

LOW DEMAND PERIOD

STATE	2/HR	4/HR	6/HR	8/HR	10/HR	12/HR	14/HR	16/HR	18/HR	20/HR
0	0.7262	0.5273	0.3791	0.2707	0.1858	0.1200	0.0750	0.0466	0.0265	0.0111
1	0.2324	0.3375	0.3677	0.3537	0.3127	0.2544	0.1943	0.1428	0.0963	0.0501
2	0.0372	0.1080	0.1783	0.2311	0.2632	0.2697	0.2516	0.2190	0.1748	0.1127
3	0.0040	0.0230	0.0577	0.1007	0.1477	0.1906	0.2172	0.2239	0.2005	0.1691
4	0.0003	0.0037	0.0140	0.0329	0.0621	0.1010	0.1407	0.1717	0.1919	0.1903
5	0.0000	0.0005	0.0027	0.0086	0.0209	0.0428	0.0729	0.1053	0.1393	0.1712
6	0.0000	0.0001	0.0004	0.0019	0.0059	0.0151	0.0315	0.0538	0.0843	0.1284
7	0.0000	0.0000	0.0001	0.0003	0.0014	0.0046	0.0116	0.0236	0.0437	0.0826
8	0.0000	0.0000	0.0000	0.0001	0.0003	0.0012	0.0038	0.0090	0.0198	0.0465
9	0.0000	0.0000	0.0000	0.0000	0.0001	0.0003	0.0011	0.0031	0.0080	0.0232
10	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0003	0.0009	0.0029	0.0105
11	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0003	0.0010	0.0043

Figure 21. Computed Erlang Formula State Probabilities

HIGH DEMAND PERIOD

STATE	2/HR	4/HR	6/HR	8/HR	10/HR	12/HR	14/HR	16/HR	18/HR	20/HR
0	.6646	.4238	.2894	.1890	.1313	.0815	.0243	.0175	.0069	.0006
1	.2694	.3723	.3626	.3297	.2446	.1919	.0907	.0680	.0272	.0027
2	.0613	.1726	.2275	.2577	.2603	.2351	.1472	.1306	.0747	.0092
3	.0047	.0262	.0885	.1330	.1926	.1943	.2139	.1672	.0939	.0113
4	+E+00	.0047	.0239	.0602	.1067	.1162	.1733	.1839	.1168	.1069
5	+E+00	.0003	.0055	.0179	.0380	.0886	.1278	.1332	.1251	.0294
6	+E+00	+E+00	.0025	.0077	.0219	.0458	.0956	.0125	.1210	.0567
7	+E+00	+E+00	.0001	.0042	.0041	.0143	.0581	.0801	.1101	.0566
8	+E+00	+E+00	+E+00	.0005	.0005	.0159	.0434	.0471	.0942	.0758
9	+E+00	+E+00	+E+00	+E=00	+E+00	.0084	.0144	.0407	.0777	.1293
10	+E+00	+E+00	+E+00	=E+00	+F+00	.0061	.0083	.0223	.0810	.2523
11	+E+00	+E+00	+E+00	+E+00	+E+00	.0020	.0031	.0068	.0715	.3593

LOW DEMAND PERIOD

STATE	2/HR	4/HR	6/HR	8/HR	10/HR	12/HR	14/HR	16/HR	18/HR	20/HR
0	.7237	.5256	.3822	.2766	.1933	.1698	.0645	.0595	.0499	.0297
1	.2340	.3533	.3845	.3656	.3282	.2792	.2092	.1701	.1063	.0963
2	.0303	.1083	.1628	.2249	.2726	.2497	.2497	.2613	.2118	.1707
3	.0030	.0129	.0531	.0962	.1333	.1608	.2367	.2221	.2176	.1810
4	+E+00	+E+00	.0141	.0236	.0560	.0763	.1320	.1414	.1483	.1532
5	+E+00	+E+00	.0021	.0087	.0101	.0494	.0758	.1163	.1163	.1253
6	+E+00	+E+00	.0011	.0033	.0053	.0181	.0360	.0421	.0784	.0954
7	+E+00	+E+00	.0001	.0008	.0013	.0070	.0145	.0196	.0519	.0653
8	+E+00	+E+00	+E+00	.0004	+E+00	.0010	.0079	.0057	.0113	.0418
9	+E+00	+E+00	+E+00	+E+00	+E+00	+E+00	.0002	.0022	.0063	.0295
10	+E+00	+E+00	+E+00	+E+00	+E+00	+E+00	+E+00	.0001	.0012	.0075
11	+E+00	+E+00	+E+00	+E+00	+E+00	+E+00	+E+00	+E+00	.0007	.0043

Figure. 22 . Simulation Program State Probabilities

APPENDIX A

COMPUTER PROGRAM TO CALCULATE ERLANG FORMULA STATE PROBABILITIES

```
      DIMENSION PROB (12,10), Serve (10)
      ZL = .0001
      NN = 1
 5    READ (5,10) (SERVE(I),I = 1,10)
10    FORMAT(IF10.5)
      LINE = 12
      DO 60 J = 1,10
      RHO = 2.0 * J/SERVE(J)
      DENOM = 0.0
      DO 30 K = 1,LINE
      FACT = 1.0
      DO 20 L = 1,K
      FACT = FACT * (L-1)
      IF (L .EQ.1) FACT = 1.0
20    CONTINUE
      DO 50 I = 1,LINE
      FACTNU = 1.0
      DO 40 L = 1.I
      FACTNU = FACTNU * (L-1)
      IF(L .EG. 1) FACTNU = 1.0
40    CONTINUE
      PROB (I,J) = (RHO**(I-1)/FACTNU)/DENOM
      IF (PROB(I,J) .LT. ZL) PROB(I,J) = 0.0
50    CONTINUE
60    CONTINUE
      WRITE (6,90)
      DO 70 I = 1, LINE
      M = I-1
      WRITE(6,65) M, (PROB(I,J), J = 1,10)
65    FORMAT (2X,I2,10F11.6)
70    CONTINUE
      IF (NN.EQ.2) STOP
      NN = 2
      GO TO 5
90    FORMAT ('1)
      END
```

APPENDIX B

SIMULATION PROGRAM: HIGH DEMAND STATE

```
PREAMBLE
      NORMALLY MODE IS INTEGER
      EVENT NOTICES INCLUDE HIGH.PRI.JOB, OPARS.JOB, RETRY,
      INPUT. QUEUE,
      TEMPORARY ENTITIES
      EVERY JOB HAS A TIME.OF.ENTRY AND A PRIORITY AND MAY
      BELONG TO THE QUEUE
      DEFINE QUEUE AS A FIFO SET RANKED BY PRIORITY
      THE SYSTEM OWNS THE QUEUE
      DEFINE IRG, THRUPUT, QUEUE.TIME AND TIME.OF ENTRY AS
      REAL VARIABLES
      DEFINE NO.REQUESTS, NO.ATTEMPTS, PRIORITY, LINES.
      BUSY, RUN, NO.BATCH, SYSTEM, BUSY AND EMPTY AS VARIABLES
      DEFINE SUMMARY AS A 2-DIMENSIONAL, REAL ARRAY
      ACCUMULATE STATE.PROB(0 to 11 BY 1) AS THE HISTOGRAM OF
      LINES,BUSY
      ACCUMULATE QUEUE.STATS(0 TO 11 BY 1) AS THE HISTOGRAM
      OF N.QUEUE
      TALLY AVE.QUEUE.TIME  AS THE AVERAGE OF QUEUE.TIME

MAIN
      RESERVE SUMMARY(*,*) AS 12 BY 10
      LET RUN = 1
      LET BUSY = 1
      SCHEDULE AN OPARS.JOB NOW
      SCHEDULE A STATISTICS IN  2880 MINUTES
      START SIMULATION
END

EVENT OPARS.JOB
      SCHEDULE AN OPARS.JOB IN EXPONENTIAL.F(30./RUN,4) MINUTES
      ADD 1 TO NO.ATTEMPTS
      IF LINES.BUSY  EQUALS 11
      SCHEDULE A RETRY IN 10 MINUTES
      RETURN
      OTHERWISE
      ADD 1 TO LINES.BUSY
      ADD 1 TO NO.REQUESTS
      LET IRG = NORMAL.F(5.1,1.0,5)
      SCHEDULE AN INPUT.QUEUE IN IRG MINUTES
      RETURN
END

EVENT HIGH.PRI.JCB
      SCHEDULE A HIGH.PRI.JOB IN EXPONENTIAL.F(17.4,3) MINUTES
      CREATE A JOB
      LET PRIORITY (JOB) = 2
      FILE JOB IN QUEUE
      RETURN
END
```

```
EVENT RETRY
      IF LINES. BUSY EQUALS 11
      ADD 1 to NO.ATTEMPTS
      SCHEDULE A RETRY IN 10 MINUTES
      RETURN
      OTHERWISE
      ADD 1 TO NO.REQUEST
      ADD 1 TO LINES.BUSY
      LET IRG = NORMA.F(5.1,1.0,5)
      SCHEDULE AN INPUT.QUEUE INIRG MINUTES
      RETURN
END

EVENT INPUT.QUEUE
      IF SYSTEM EQUALS EMPTY AND NO.BATCH IS LESS THAN 2
      SCHEDULE A JOB.COMPLETION IN GAMMA.F(4.43,3.1,7)MINUTES
      ADD 1 TO NO.BATCH
      OTHERWISE
      CREATE A JOB
      LET TIME.OF.ENTRY(JOB) = TIME.V
      LET PRIORITY(JOB) = 1
      FILE JOB IN QUEUE
      REGARDLESS
      RETURN
END

EVENT EXECUTION
      IF SYSTEM EQUALS BUSY
      SCHEDULE AN EXECUTUION IN EXPONENTIAL.F(1.12,2) MINUTES
      REGARDLESS
      IF N.QUEUE EQUALS 0
      RETURN
      OTHERWISE
      IF PRIORITY(F.QUEUE) = 2
      REMOVE FIRST JOB FROM QUEUE
      DESTROY THE JOB
      RETURN
      OTHERWISE
      IF NO.BATCH  EQUALS 2
      RETURN
      OTHERWISE
      REMOVE FIRST JOB FROM QUEUE
      LET QUEUE.TIME= (TIME.V -TIME.OF.ENTRY(JOB) * 1440.
      DESTROY THE JOB
      IF SYSTEM EQUALS BUSY
      SCHEDULE A JOB.COMPLETION IN GAMMA.F(6.06,4.34,6)MINUTES
      OTHERWISE
      SCHEDULE A JOB.COMPLETION IN GAMMA.F(4.43,3.1,7)MINUTES
      REGARDLESS
      ADD 1 TO NO.BATCH
      RETURN
END
```

```
EVENT JOB.COMPLETION
       SUBTRACT 1 FROM LINES.BUSY
       SUBTRACT 1 FROM NO.BATCH
       IF N.QUEUE EQUALS 0
       RETURN
       OTHERWISE
       IF SYSTEM EQUALS BUSY
       CANCEL THE EXECUTION
       REGARDLESS
       SCHEDULE AN EXECUTION NOW
       RETURN
END

EVENT STATISTICS
      START NEW PAGE
      PRINT 1 LINE WITH RUN*2 THUS
ARRIVAL RATE = **HOUR
      SKIP 3 OUTPUT LINES
      PRINT 1 LINE THUS
      PR(BUSY LINES)   PR(INPUT QUEUE)
      SKIP 2 OUTPUT LINES
      FOR I = 1 TO 12, DO
      PRINT 1 LINE WITH I-1, STATE.PROB(I)/2 AND QUEUE.
      STATS(I)/2 THUS
**       *****         *****
      LOOP
      SKIP 5 OUTPUT LINES
      PRINT 3 LINES WITH AVE.QUEUE.TIME,NO.ATTEMPTS AND NO.
      REQUESTS THUS
AVERAGE OPARS QUEUE TIME = **.****
NO.OPARS  ATTEMPTED   ****
NO.OPARS  COMPLETE ****
      FOR I = 1 to 12, DO
      LET SUMMARY(I,RUN) = STATE.PROB(I)/2
      LOOP
      SCHEDULE A STATISTICS IN 2880 MINUTES
      RESET TOTALS OF LINES.BUSY, N.QUEUE AND QUEUE.TIME
      IF RUN EQUALS 10
      GO TO FINAL.STATS
      OTHERWISE
      ADD 1 TO RUN
      LET NO.ATTEMPTS  = 0
      LET NO.REQUESTS  = 0
      RETURN
      'FINAL.STATS'
      START NEW PAGE
      PRINT 5 LINES THUS
EMPTY SYSTEM SUMMARY
STATE  2/HR 4/HR 6/HR  8/HR  10/HR  12/HR 14/HR  16/HR  18/HR  20/HR
      FOR I = 1 to 12, DO
      PRINT 1 LINE WITH I-1, SUMMARY (I,1), SUMMARY (I,2),
      SUMMARY(I,3), SUMMARY (I,4),SUMMARY(I,5),SUMMARY(I,6),
      SUMMARY(I,7), SUMMARY (I,8),SUMMARY(I,9),SUMMARY(I,10),THUS
** .**** .**** .**** .**** .**** .**** .**** .**** .**** .****
LOOP
STOP
END
```

APPENDIX C

SIMULATION PROGRAM: LOW DEMAND

```
PREAMBLE
      NORMALLY MODE IS INTEGER
      EVENT NOTICES INCLUDE HIGH.PRT.JOB,OPARS.JOB,RETRY,INPUT.
      QUEUE,
      TEMPORARY ENTITIES
      EVERY JOB HAS A TIME.OF.ENTRY AND A PRIORITY AND MAY
      BELONG TO THE QUEUE
      DEFINE QUEUE AS A FIFO SET RANKED BY PRIORITY
      THE SYSTEM OWNS THE QUEUE
      DEFINE IRG,THRUPUT,QUEUE.TIME AND TIME.OF.ENTRY AS
      REAL VARIABLES
      DEFINE NO.REQUESTS; NO.ATTEMPTS,PRIORITY, LINES.BUSY,
      RUN, NO.BATCH, SYSTEM, BUSY AND EMPTY AS VARIABLES
      DEFINE SUMMARY AS A 2-DIMENSIONAL, REAL ARRAY
      ACCUMULATE STATE.PROB(0 TO 11 BY 1) AS THE HISTOGRAM
      OF LINES.BUSY
      ACCUMULATE QUEUE.STATS.(0 TO 11 BY 1) AS THE HISTOGRAM
      OF N.QUEUE TALLY AVE.QUEUE.TIME AS THE AVERAGE OF QUEUE.
      TIME
END

MAIN
      RESERVE SUMMARY(*,*) AS 12 BY 10
      LET RUN = 1
      LET SYSTEM = 1
      LET BUSY = 1
      SCHEDULE AN OPARS.JOB NOW
      SCHEDULE AN EXECUTION IN EXPONENTIAL.F(1,12,2)MINUTES
      SCHEDULE A HIGH.PRI.JOB IN EXPONENTIAL.F(17.4,3)MINUTES
      CREATE A JOB
      LET PRIORITY (JOB) = 2
      FILE JOB IN QUEUE
      SCHEDULE A STATISTICS IN 2880 MINUTES
      START SIMULATION
END

EVENT OPARS.JOB
      SCHEDULE AND OPARS.JOB IN EXPONENTIAL.F(30./RUN,4) MINUTES
      ADD 1 TO NO.ATTEMPTS
      IF LINES.BUSY EQUALS 11
      SCHEDULE A RETRY IN 10 MINUTES
      RETURN
      OTHERWISE
      ADD 1 TO LINES.BUSY
      ADD 1 TO NO. REQUESTS
      LET IRG = NORMA.F(5.1,1.0,5)
      SCHEDULE AN INPUT.QUEUE IN IRG MINUTES
      RETURN
END

EVENT HIGH.PRI.JOB
      SCHEDULE A HIG.PRI.JOB IN EXPONENTIAL.F(17.4,3)MINUTES
      CREATE A JOB
```

APPENDIX C

SIMULATION PROGRAM: LOW DEMAND STATE CONT'D

```
     LET PRIORITY(JOB) = 2
     FILE JOB IN QUEUE
     RETURN
END

EVENT RETRY
     IF LINES.BUSY EQUALS 11
     ADD 1 TO NO.ATTEMPTS
     SCHEDULE A RETRY IN 10 MINUTES
     RETURN
     OTHERWISE
     ADD 1 TO NO.REQUESTS
     ADD 1 TO LINES.BUSY
     LET IRG = NORMAL.F(5.1,1.0,5)
     SCHEDULE AN INPUT.QUEUE IN IRG MINUTES
     RETURN
END

EVENT INPUT.QUEUE
     IF SYSTEM EQUALS EMPTY AND NO.BATCH IS LESS THAN 2
     SCHEDULE A JOB.COMPLETION IN GAMMA.F(4.43,3.1,7) MINUTES
     ADD 1 TO NO.BATCH
     OTHERWISE
     CREATE A JOB
     LET TIME.OF.ENTRY(JOB) = TIME.V
     LET PRIORITY(JOB) = 1
     FILE JOB IN QUEUE
     REGARDLESS
     RETURN
END

EVENT EXECUTION
     IF SYSTEM EQUALS BUSY
     SCHEDULE AN EXECUTION IN EXPONENTIAL.F(1.12,2) MINUTES
     REGARDLESS
     IF N.QUEUE EQUALS C
     RETURN
     OTHERWISE
     IF PRIORITY (F.QUEUE) = 2
     REMOVE FIRST JOB FROM QUEUE
     DESTORY THE JOB
     RETURN
     OTHERWISE
     IF NO.BATCH EQUALS 2
     RETURN
     OTHERWISE
     REMOVE FIRST JOB FROM QUEUE
     LET QUEUE.TIME = (TIME.V - TIME.OF.ENTRY(JOB)) * 1440.
     DESTORY THE JOB
     IF SYSTEM EQUALS BUSY
     SCHEDULE A JOB.COMPLETION IN GAMMA.F(6.06,4.34,6)MINUTES
```

SIMULATION PROGRAM: LOW DEMAND STATE CONT'D

```
     OTHERWISE
     SCHEDULE A JOB.COMPLETION IN GAMMA.F(4.43,3.1,7) MINUTES
     REGARDLESS
     ADD 1 TO NO.BATCH
     RETURN
END

EVENT JOB.COMPLETION
     SUBTRACT 1 FROM LINES.BUSY
     SUBTRACT 1 FROM NO.BATCH
     IF N.QUEUE EQUALS 0
     RETURN
     OTHERWISE
     IF SYSTEM EQUALS BUSY
     CANCEL THE EXECUTION
     REGARDLESS
     SCHEDULE AND EXECUTION NOW
     RETURN
END

EVENT STATISTICS
     START NEW PAGE
     PRINT 1 LINE WITH RUN*2 THUS
ARRIVAL RATE = **/HOUR
     SKIP 3 OUTPUT LINES
     PRINT 1 LINE THUS
     PR(BUSY LINES)  PR(INPUT LINES)
     SKIP 2 OUTPUT LINES
     FOR I = 1 TO 12, DO
     PRINT 1 LINE WITH I-1, STATE.PROBLE(I).2 AND QUEUE.STATS
     (I)/2 THUS
**          .*****          .*****
     LOOP
     SKIP 5 OUTPUT LINES
     PRINT 3 LINES WITH AVE.QUEUE.TIME,NO.ATTEMPTS AND NO.
     REQUESTS THUS
AVERAGEOPARS QUEUE TIME = **.****
NO.OPARS   ATTEMPTED  ****
NO.OPARS   COMPLETED  ****
     FOR I = 1 TO 12, DO
     LET SUMMARY(I,RUN) = STATE.PROB(I)/2
     LOOP
     SCHEDULE A STATISTICS IN 2880 MINUTES
     RESET TOTALS OF LINES.BUSY,N.QUEUE AND QUEUE.TIME
     IF RUN EQUALS 10
     GO TO FINAL.STATS
     OTHERWISE
     ADD 1 TO RUN
     LET NO.ATTEMPTS = 0
     LET NO.REQUESTS  = 0
     RETURN
     'FINAL.STATS'
     START NEW PAGE
     PRINT 5 LINES THUS
BUSY SYSTEM SUMMARY
```

```
STATE 2/HR 4/HR 6/HR 8/HR 10/HR 12/HR 14/HR 16/HR 18/HR 20/HR

      FOR I = 1 TO 12, DC
       PRINT 1 LINE WITH I-1, SUMMARY (I,1), SUMMARY (I,2),
       SUMMARY (I,3),SUMMARY(I,4),SUMMARY(I,5),SUMMARY(I,6),
       SUMMARY (I,7), SUMMARY(I,8), SUMMARY (I,9), SUMMARY
       (I,10) THUS
** .**** .**** .**** .**** .**** .**** .**** .**** .****.****

LOOP
STOP
END
```

REFERENCES

1. Gaver, D.P. and Thompson, G.L., Programming and Probability Models in Operations Research, p. 452-521 Brooks/Cole, 1973.

2. Giffin, W.C., Queueing, p.241-252,Grid,1973.

3. Gross, D. and Harris, C.M., Fundamentals of Queueing Theory, p. 109-111, Wiley, 1974.

4. Hillier, F.S. and Leiberman, G.J., Introduction to Operations Research, P.419, Holden-Day, 1974.

5. Kleinrock, L., Queueing Systems, V.1, Wiley, 1975.

6. Mood, A.M., Graybill, F.A. and Boes, D.C., Introduction to the Theory of Statistics, 3rd ed., McGraw-Hill, 1974.

7. Thorton, J.E., Design of a Computer the Control Data 6600, Scott, Foresman, 1970.

INITIAL DISTRIBUTION LIST

Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
Department Chairman, Code 55 Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
Professor D.P. Gaver, Code 55Gv Naval Postgraduate School Monterey, California 93940	3
Lieutenant E. Roland Code 55Ro Naval Postgraduate School Monterey, California 93940	1
Lieutenant Kenneth L. Smith, USN RD #4 Box 217 Johnstown, Pennsylvania 15905	1